Robust Verification of Concurrent Stochastic Games

Angel Y. He and David Parker

Department of Computer Science, University of Oxford, Oxford OX1 2JD, UK
angel.he@balliol.ox.ac.uk, david.parker@cs.ox.ac.uk

Abstract. Autonomous systems often operate in multi-agent settings and need to make concurrent, strategic decisions, typically in uncertain environments. Verification and control problems for these systems can be tackled with concurrent stochastic games (CSGs), but this model requires transition probabilities to be precisely specified — an unrealistic requirement in many real-world settings. We introduce *robust CSGs* and their subclass *interval CSGs* (ICSGs), which capture epistemic uncertainty about transition probabilities in CSGs. We propose a novel framework for *robust* verification of these models under worst-case assumptions about transition uncertainty. Specifically, we develop the underlying theoretical foundations and efficient algorithms, for finite- and infinite-horizon objectives in both zero-sum and nonzero-sum settings, the latter based on (social-welfare optimal) Nash equilibria. We build an implementation in the PRISM-games model checker and demonstrate the feasibility of robust verification of ICSGs across a selection of large benchmarks.

Keywords: Robust quantitative verification · Probabilistic model checking · Concurrent stochastic games · Epistemic uncertainty.

1 Introduction

Autonomous and intelligent systems are increasingly deployed in environments that are *nondeterministic*, *stochastic* and *concurrent*, such as autonomous vehicle coordination, robotic exploration and networked market interactions. In these settings, decision-making involves simultaneous strategic interactions between multiple agents, often within uncertain and dynamic environments.

Concurrent stochastic games (CSGs) [47], also known as *Markov games*, provide a powerful framework for modelling such multi-agent systems. Unlike the simpler model of *turn-based* stochastic games (TSGs) [15], CSGs allow players to select their actions simultaneously, without knowledge of each other's choices. The outcomes depend probabilistically on the players' joint actions.

Formal verification techniques for CSGs provide a means to establish quantitative guarantees on the behaviour of these stochastic multi-agent systems, e.g., ensuring that "a drone can safely reach its target with at least 95% probability, regardless of the actions of other aircraft". They can also be used to automatically synthesise controllers or strategies that achieve these guarantees. Early

S. Junges and G. Katz (Eds.): TACAS 2026, LNCS 16505, pp. 505–525, 2026.
https://doi.org/10.1007/978-3-032-22752-2_26

work on these models focused on the zero-sum setting (e.g., [9, 17, 18]), whilst more recent work has added support for various temporal logics and the use of nonzero-sum game-theory solution concepts such as Nash equilibria (NE) and their variants [34, 35], along with widely used tool support [31].

Despite their modelling effectiveness, a limitation of CSGs is that they assume transition probabilities are *precisely known*. In reality, system dynamics are often only partially known due to abstraction, modelling inaccuracies, noise, or limited data in learned statistical models. This is particularly evident in data-driven contexts like (model-based) reinforcement learning (RL), where transition probabilities are estimated from data. These issues limit the reliability of guarantees from verification and can make synthesised strategies sub-optimal.

In recent years, there has been growing interest in principled approaches to reasoning about *epistemic uncertainty* in probabilistic models for verification [3]. For the simpler, single-agent setting, where decision making is performed using Markov decision processes (MDPs), a well studied approach is *robust MDPs* (RMDPs) [28, 41, 54], which capture model uncertainty via a set of possible transition probability functions. A common subclass is *interval MDPs* (IMDPs) [22], where transition probabilities are bounded within intervals. Verification techniques then provide guarantees or synthesise optimal controllers in a *robust* manner, i.e., making *worst-case* assumptions about model uncertainty.

However, analogous frameworks for stochastic multi-agent systems remain underdeveloped. In this paper, we address that gap and propose *robust concurrent stochastic games* (RCSGs), a novel verification framework that augments CSGs with transition uncertainty and robust solution concepts. In fact, RMDPs already have a link to stochastic games: they can be interpreted as a *zero-sum TSG* between the agent and an adversarial *nature* player that resolves uncertain transition probabilities; this view underlies many algorithms for solving RMDPs [10, 28, 39, 41]. However, extending this framework to *multi-agent*, and especially *concurrent* settings, is non-trivial, as the interplay between transition uncertainty and simultaneous player actions significantly complicates both the reasoning and the very definition of robustness.

Contributions and challenges. In this work, we develop a framework for robust verification of CSGs under adversarial transition uncertainty, covering both zero-sum and nonzero-sum settings with finite- and infinite-horizon objectives. We focus primarily on the subclass of interval CSGs (ICSGs) characterised by transition probability intervals. Extending robustness from MDPs to concurrent multi-agent games introduces fundamental challenges: optimality requires mixed strategies; equilibria definitions must incorporate uncertainty resolutions; and, in the nonzero-sum case, the adversarial role of nature differs from standard best-response reasoning. To address these, we: introduce robust equilibrium notions; establish theoretical results, e.g., on value preservation under player/nature action ordering; and present novel reductions from ICSGs to (non-robust) CSGs by adding an adversarial nature player. The latter yields a 2-player game in the zero-sum case, and a more subtle 3-player construction in the nonzero-sum case where nature minimises social welfare. Building on these results, we derive tractable algorithms for solving ICSGs and implement them in PRISM-games [31]. We show their practicality via empirical evaluations on a set of large benchmarks: verification for zero-sum ICSGs performs comparably to CSGs, while nonzero-sum methods scale effectively but also provide insights into the intrinsic challenges of robust multi-agent reasoning.

Related work. In the *single-agent* setting, RMDPs are solvable via robust dynamic programming (RDP) [28, 41, 54]. Recent work develops generic algorithms for polytopic [10, 53] and more general RMDPs with constant support (e.g., [39]) via reduction to TSGs. Robust methods for multi-agent settings are more limited, restricted to *turn-based* polytopal stochastic games [8], *qualitative* verification [5]; or *sampling-based*, learning-driven algorithms (e.g., [19, 45, 48]) for the similar problem of distributionally *robust Markov games* [21, 37, 47] in RL, without model-checking capabilities or verification guarantees.

2 Preliminaries

Let $\mathcal{D}(X)$ denote the set of discrete probability distributions over a finite set X, and let $\mathbb{1}[A]$ be the indicator function that equals 1 if A holds and 0 otherwise.

2.1 Robust Markov Decision Processes

A core model for verification and control tasks in the context of uncertainty is *Markov decision processes (MDPs)* [4, 27].

Definition 1 (MDP). *A* Markov decision process *(MDP) is a tuple $M = (S, \bar{s}, A, P)$, where S is a finite set of states with initial state $\bar{s} \in S$; A is a finite set of actions; and $P : S \times A \rightharpoonup \mathcal{D}(S)$ is a probabilistic transition function.*

$A(s)$ denotes the enabled actions in state s. We write $P_{sa} = P(s, a)$ for the next-state distribution at s under action a and $P_{sas'} = P_{sa}(s')$ for the corresponding transition probability to s'. A *path* is a finite or infinite sequence $\pi = s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} \dots$ such that $s_0 = \bar{s}$, $a_i \in A(s_i)$ and $P_{s_i a_i s_{i+1}} > 0$ for all i. We write $\pi(i) = s_i$, $\pi[i] = a_i$, and let $FPaths_M$ and $IPaths_M$ be the sets of finite and infinite paths in M, respectively.

A *strategy* (or policy) of M is a function $\sigma : FPaths_M \to \mathcal{D}(A)$ that resolves the choices of action in each state. Typically, we aim to find an optimal strategy for an MDP, e.g., one that maximises the probability of a target state set being reached or the expected value of some reward function.

In order to reason about MDPs *robustly* in the context of (epistemic) uncertainty about the model itself, we can use *robust MDPs.*

Definition 2 (RMDP). *A* robust MDP *(RMDP) is a tuple $M_R = (S, \bar{s}, A, \mathcal{P})$, where S, $\bar{s}$ and A are as for MDPs (Definition 1), and $\mathcal{P} : S \times A \rightharpoonup 2^{\mathcal{D}(S)}$ is an uncertain probabilistic transition function.*

Intuitively, an RMDP captures unknown transition dynamics: for each state s and action $a \in A(s)$, the *uncertainty set* $\mathcal{P}_{sa} = \mathcal{P}(s, a)$ represents the *set* of possible next-state distributions. Selecting a single $P_{sa} \in \mathcal{P}_{sa}$ for each (s, a) yields a probabilistic transition function $P : S \times A \to \mathcal{D}(S)$, giving a specific MDP. Abusing notation slightly, we also treat $\mathcal{P}$ as a set and write $P \in \mathcal{P}$, referring to each P as an *uncertainty resolution*. Typically, we aim to find a *robust optimal* strategy, i.e., one that is optimal against the worst-case uncertainty resolution.

An RMDP M_R can be viewed as a zero-sum TSG, i.e., a game which alternates between an agent choosing an $a \in A(s)$ in each state s and then an adversarial player *nature* resolving the choices $P_{sa} \in \mathcal{P}_{sa}$. Assumptions or restrictions on the strategies for nature dictate the kind of uncertainty considered: 1) *rectangularity* [28,41], i.e., whether transition uncertainty is resolved independently across different states (*s-rectangular*) or state-action pairs (*(s, a)-rectangular*); 2) *static* (*stationary*) vs. *dynamic* (*time-varying*) semantics [28,41], i.e., whether nature follows a *memoryless* strategy that has to make consistent choices at each (s, a) over time. We can also restrict the nature of the uncertainty sets $\mathcal{P}_{sa}$, notably whether they are polytopic, i.e., next-state distributions in $\mathcal{P}_{sa}$ form a polytope. In this work, we focus on (s, a)-rectangular, polytopic uncertainty. This setting includes the well-studied class of IMDPs [22,41], where each $\mathcal{P}_{sa}$ is defined by independent intervals over transition probabilities.

2.2 Concurrent Stochastic Games

CSGs [47] provide the semantic basis for the class of games we introduce.

Definition 3 (CSG). *An (n-player)* concurrent stochastic game *(CSG) is a tuple* $\mathcal{G} = (N, S, \bar{s}, A, \Delta, P)$ *where* $N = \{1, \ldots, n\}$ *is a finite set of players;* S, $\bar{s} \in S$ *and* $P : S \times A \to \mathcal{D}(S)$ *are as defined for an MDP (Definition 1);* $A = \times_{i \in N}(A_i \cup \{\bot\})$ *where* A_i *is the set of actions for player* i *and* $\bot$ *is an idle action disjoint from* $\cup_{i \in N} A_i$*;* $\Delta : S \to 2^{\cup_{i \in N} A_i}$ *is an action assignment function.*

A CSG $\mathcal{G}$ begins in the initial state $\bar{s}$. At each state $s \in S$, each player $i \in N$ simultaneously selects an action $a_i \in A_i(s)$, where $A_i(s) = \Delta(s) \cap A_i$ if $\Delta(s) \cap A_i \neq \emptyset$ and $A_i(s) = \{\bot\}$ otherwise. The game then transitions to state s' following the distribution P_{sa}, where $a = (a_1, \ldots, a_n) \in A(s) := \times_{i \in N} A_i(s)$.

To allow quantitative analysis of $\mathcal{G}$, we augment CSGs with *reward structures*.

Definition 4 (Reward structure). *A* reward structure *for a CSG* $\mathcal{G}$ *is a tuple* $r = (r_A, r_S)$ *where* $r_A : S \times A \to \mathbb{R}$ *is the* action reward *function, and* $r_S : S \to \mathbb{R}$ *is the* state reward *function. We denote the total reward associated with a state-action pair* (s, a) *as* $r_{sa} = r(s, a) := r_A(s, a) + r_S(s)$*.*

A *strategy* for player i is a function $\sigma_i : FPaths_{\mathcal{G}} \to \mathcal{D}(A_i)$ mapping finite histories to distributions over actions. A *strategy profile* (or just *profile*) is a tuple of strategies for each player, denoted $\sigma = (\sigma_1, \ldots, \sigma_n) \in \Sigma := \times_{i \in N} \Sigma_i$. An *objective* (or utility function) of player i is a random variable $X_i : IPaths_{\mathcal{G}} \to \mathbb{R}$. In a *zero-sum* game, which is 2-player by definition, players have directly opposing objectives, i.e., $X_1 = -X_2$. In this case, we will represent their objectives using a single variable $X := X_1$, so that $X_2 = -X$. In the *nonzero-sum* (or *general-sum*) case, we write $X = (X_1, \ldots, X_n)$ for the tuple of all player objectives.

In this paper we focus on the four common objectives below, two of which are finite-horizon and two infinite-horizon. We assume a set of target states $T \subseteq S$ and, for the finite-horizon case, a time horizon $k \in \mathbb{N}$.

- *Bounded probabilistic reachability*: $X(\pi) = \mathbb{1}\,[\exists j \leq k.\ \pi(j) \in T]$;
- *Bounded cumulative reward*: $X(\pi) = \sum_{i=0}^{k-1} r(\pi(i), \pi[i])$;
- *Probabilistic reachability*: $X(\pi) = \mathbb{1}\,[\exists j \in \mathbb{N}.\ \pi(j) \in T]$; and
- *Reachability reward*: $X(\pi) = \sum_{i=0}^{k_{\min}-1} r(\pi(i), \pi[i])$ if $\exists j \in \mathbb{N}.\ \pi(j) \in T$ and $X(\pi) = \infty$ otherwise, where $k_{\min} = \min\{j \in \mathbb{N} \mid \pi(j) \in T\}$.

We denote the *expected utility* of player i from state s under profile σ in $\mathcal{G}$ as $u_i(\sigma \mid s, X) := V^i_{\mathcal{G}}(s \mid \sigma, X) := \mathbb{E}^{\sigma}_{\mathcal{G},s}[X_i]$, with the index i omitted in the zero-sum case. In zero-sum games, the *value* of $\mathcal{G}$ with respect to X exists if the game is *determined*, i.e., the maximum payoff that player 1 can guarantee equals the minimum payoff player 2 can enforce; the corresponding strategies are said to be *optimal*. For nonzero-sum games where players may cooperate or compete, we use the concept of a *Nash equilibrium* (NE): a profile in which no player can improve their utility by unilaterally deviating. A *social-welfare optimal NE* (SWNE) [34] refers to an NE that also maximises the players' total utility.

A special, degenerate "one-shot" case of a CSG is a *normal form game* (NFG), which consists of a single state and a single decision round. Thus, an NFG can be represented as a simplified tuple $\mathcal{Z} = (N, A, u)$, where N and A are as defined for a CSG, and $u = (u_1, \ldots, u_n)$ with $u_i : A \to \mathbb{R}$ defining player i's utility for each joint action. A 2-player NFG can be represented as a *bimatrix game*, defined by two matrices $\mathcal{Z}_1, \mathcal{Z}_2 \in \mathbb{R}^{l \times m}$ with entries $z^1_{ij} = u_1(a_i, b_j)$ and $z^2_{ij} = u_2(a_i, b_j)$, where $A_1 = \{a_1, \ldots, a_l\}$ and $A_2 = \{b_1, \ldots, b_m\}$. The game is called *zero-sum* if $\forall a \in A.\ u_1(a) + u_2(a) = 0$, in which case we can represent it as a single *matrix game* $\mathcal{Z} \in \mathbb{R}^{l \times m}$ with $z_{ij} = u_1(a_i, b_j) = -u_2(a_i, b_j)$, i.e., $\mathcal{Z} = \mathcal{Z}_1 = -\mathcal{Z}_2$.

3 Robust Concurrent Stochastic Games

We now propose the model of *robust CSGs* (RCSGs), which unifies the notions of *robustness* from RMDPs and *concurrent* decision-making from CSGs.

Definition 5 (RCSG). *A* robust CSG *(RCSG) is a tuple* $\mathcal{G} = (N, S, \bar{s}, A, \Delta, \mathcal{P})$ *where* $\mathcal{P}$ *is an* uncertain *transition function defined as for RMDPs in Definition 2, and all other components are as defined for CSGs in Definition 3.*

Similar to the way that fixing the transition function in an RMDP induces an MDP, fixing the transition function in an RCSG to a particular $P \in \mathcal{P}$ induces a CSG $\mathcal{G}_P = (N, S, \bar{s}, A, \Delta, P)$. We parametrise the corresponding notation with P. Notably, for a state s of $\mathcal{G}$ and a strategy profile σ (defined as for CSGs), we write $u_i(\sigma, P \mid s, X) := \mathbb{E}^{\sigma,P}_{\mathcal{G},s}[X]$[1] for the expected value of an objective X under σ applied to $\mathcal{G}_P$.

[1] We use these interchangeably and omit parameters that are clear from the context.

By contrast to the uncertainty semantics in RMDPs (see Section 2.1), multi-player RCSGs introduce an additional dimension: whether uncertainty is resolved *adversarially* or is *controlled* by players [8]. In the adversarial case, nature resolves uncertainty against the players: in zero-sum games, where players have directly opposing objectives, nature aligns with one player to minimise the other's payoff; in nonzero-sum games, it acts against both by minimising a joint objective such as social welfare or cost.

The controlled case assumes that one or more players resolve uncertainty to optimise their own objectives. This corresponds to optimistic reasoning in single-agent or zero-sum settings and can be seen as the dual of the adversarial case. However, in nonzero-sum games, assigning control to a single player can undermine fairness by attributing uncertainty to that player's decisions, while shared control would require principled coordination among players. We therefore focus on the adversarial resolution in both zero- and nonzero-sum games.

Next, we define the *robust* analogue of several game-theoretic concepts in the context of RCSGs. In general, we enforce that their defining properties hold under every $P \in \mathcal{P}$, or equivalently under the worst case $P^* := \arg\min_{P\in\mathcal{P}} \mathbb{E}^{\sigma,P}_{\mathcal{G},s}[X]$.

Zero-sum RCSGs. We first adapt classical minimax concepts for (2-player) zero-sum games, assuming that player 1 maximises an objective X.

Definition 6 (Robust determinacy and optimality). *A zero-sum RCSG $\mathcal{G}$ is* robustly determined *with respect to an objective X, if for any state $s \in S$:*

$$\sup_{\sigma_1\in\Sigma_1} \inf_{\sigma_2\in\Sigma_2} \inf_{P\in\mathcal{P}} \mathbb{E}^{(\sigma_1,\sigma_2),P}_{\mathcal{G},s}[X] = \inf_{\sigma_2\in\Sigma_2} \sup_{\sigma_1\in\Sigma_1} \inf_{P\in\mathcal{P}} \mathbb{E}^{(\sigma_1,\sigma_2),P}_{\mathcal{G},s}[X] =: V_{\mathcal{G}}(s,X)$$

where we call $V_{\mathcal{G}}(s,X)$ the robust value *of $\mathcal{G}$ in s with respect to X. Also, $\sigma_1^* \in \Sigma_1$ is a* robust optimal strategy *of player 1 with respect to X if $\mathbb{E}^{(\sigma_1^*,\sigma_2),P}_{\mathcal{G},s}[X] \geq V_{\mathcal{G}}(s,X)$ for all $s \in S, \sigma_2 \in \Sigma_2, P \in \mathcal{P}$; similarly $\sigma_2^* \in \Sigma_2$ is a* robust optimal strategy *of player 2 if $\mathbb{E}^{(\sigma_1,\sigma_2^*),P}_{\mathcal{G},s}[X] \leq V_{\mathcal{G}}(s,X)$ for all $s \in S, \sigma_1 \in \Sigma_1, P \in \mathcal{P}$.*

Nonzero-sum RCSGs. In the nonzero-sum case, each player $i \in N$ has a distinct objective X_i. For this setting, we adopt the concept of a *robust Nash equilibrium* (RNE) [1,30,43], which refers to a profile σ^* that remains a Nash equilibrium under any uncertainty resolution. Note that, in the zero-sum case, RNE coincide with the notion of robust optimal strategies.

As is common for CSGs, we use *subgame-perfect* NE [42], which require equilibrium behaviour in every state of the game, not just the initial one. We call these *subgame-perfect RNE* but, for brevity, often refer to them simply as RNE. In standard CSGs with infinite-horizon objectives, NE may not exist [6], but ε-NE do exist for any $\varepsilon > 0$ under the objectives we consider. We therefore work with *subgame-perfect ε-RNE* for infinite-horizon properties.

Definition 7 (Subgame-perfect ε-RNE). *A profile σ^* is a* subgame-perfect robust ε-NE *(ε-RNE) iff $\varepsilon + \inf_{P\in\mathcal{P}} \left[u_i(\sigma^*_{-i}[\sigma^*_i], P) - u_i(\sigma^*_{-i}[\sigma_i], P)\right] \geq 0$ for all $\sigma_i \in \Sigma_i, i \in N$ at* every state $s \in S$. *We define $\langle \inf_{P\in\mathcal{P}} u_i(\sigma^*, P)\rangle_{i\in N}$ as the corresponding* ε-RNE values. *A* subgame-perfect robust NE *(RNE) is an ε-RNE* *with $\varepsilon = 0$. We write $\Sigma_{\varepsilon\text{-}RNE}$ for the set of all ε-RNE and Σ_{RNE} for the set of all RNE.*

Even if all induced CSGs $\mathcal{G}_P$ of an RCSG $\mathcal{G}$ have an NE (ε-NE), there may not exist an RNE (ε-RNE). This is because a profile that is an NE in one induced CSG may not be an NE across all others. See the extended version of this paper [24] for an illustration.

Next, we propose the robust counterparts of SWNE [34, 46] as RNE that maximise the robust (worst-case) total utility of the players, denoted $u_+(\sigma, P) := \sum_{i \in N} u_i(\sigma, P)$ for a given profile $\sigma \in \Sigma$ and $P \in \mathcal{P}$.

Definition 8 (RSWNE). *An RNE σ^* of $\mathcal{G}$ is a* robust social-welfare optimal NE *(RSWNE) if it maximises the* robust social welfare *amongst all RNE, i.e., $\sigma^* \in \arg\max_{\sigma \in \Sigma_{RNE}} u_+(\sigma, P^*_\sigma)$ where $P^*_\sigma := \arg\inf_{P \in \mathcal{P}} u_+(\sigma, P)$. We define $\langle u_i(\sigma^*, P^*_{\sigma^*}) \rangle_{i \in N}$ as the corresponding* RSWNE values*.*

Like RMDPs, various uncertainty models are applicable in RCSGs, such as those characterised by L^p-balls [26, 49] and non-rectangular sets. However, value computation is often computationally intractable under these models, even in single-agent settings [54]. By contrast, *interval* uncertainty yields convex uncertainty sets, enabling tractable computation while effectively capturing bounded but unstructured estimation errors, e.g., those derived from confidence intervals [50]. Hence, for the remainder of the paper we focus on *interval CSGs*, as a natural and scalable foundation for incorporating robustness into CSGs.

Definition 9 (ICSG). *An* interval CSG *(ICSG) is a tuple $\mathcal{G} = (N, S, \bar{s}, A, \Delta, \check{P}, \hat{P})$ where $\check{P}, \hat{P} : S \times A \times S \rightharpoonup [0, 1]$ are partial functions that assign lower and upper bounds, respectively, to transition probabilities, such that $\check{P}_{sas'} \leq \hat{P}_{sas'}$. All other components are defined as for CSGs (Definition 3).*

An ICSG is an RCSG where $\mathcal{P}_{sa} = \{P_{sa} \in \mathcal{D}(S) \mid \forall s' \in S.\ P_{sas'} \in [\check{P}_{sas'}, \hat{P}_{sas'}]\}$. We also require that $\check{P}_{sas'} = 0 \iff \hat{P}_{sas'} = 0$, i.e., each transition is either excluded or assigned a non-degenerate interval with a strictly positive lower bound. This enables the standard *graph preservation* constraint [12, 39], which requires that all $P \in \mathcal{P}$ share the same support. This property is essential for ensuring the tractability of RDP [28, 41] over (s, a)-rectangular uncertainty models.

4 Zero-sum ICSGs

We now establish the theoretical foundations for robust verification of ICSGs, starting with the *zero-sum* case. We fix an ICSG $\mathcal{G} = (N, S, \bar{s}, A, \Delta, \check{P}, \hat{P})$ where $N = \{1, 2\}$ and in which player 1 maximises an objective X. For now, we assume that X is infinite-horizon: either probabilistic/reward reachability.

At a high level, analogously to the stochastic game view of an RMDP, we will reduce ICSG $\mathcal{G}$ to a CSG $\mathcal{G}^{\mathcal{A}}$ extended with a third player, *nature*, who resolves transition uncertainty adversarially against player 1. Since player 2 and 3 (nature) share the same objective, they can be merged into a single coalition, making $\mathcal{G}^{\mathcal{A}}$ a 2-player CSG between coalitions $\{1\}$ and $\{2, 3\}$. We refer to $\mathcal{G}^{\mathcal{A}}$ as the *adversarial expansion* of $\mathcal{G}$. We will establish a one-to-one correspondence between optimal values and strategies in $\mathcal{G}^{\mathcal{A}}$ and their robust counterparts in $\mathcal{G}$, allowing us to reduce robust verification of zero-sum ICSGs to solution of zero-sum CSGs. The latter can be performed with value iteration [34] although, as for RMDPs, explicit construction of the full CSG $\mathcal{G}^{\mathcal{A}}$ is not required.

Player-first vs. nature-first semantics. In the CSG reduction, a natural question to consider is the order in which uncertainty is resolved relative to players' moves. Under the *player-first* semantics, nature acts *after* both players have chosen their actions. While this aligns closely with the adversarial interpretation of robustness (see Definition 6), it requires nature's minimisation problem (against player 1's objective) to be solved separately for every player profile $\sigma \in \Sigma$, which is computationally demanding. By contrast, the *nature-first* semantics assumes that nature first commits to a realisation of $\mathcal{P}$ *before* any player acts, thereby inducing a fixed CSG upfront. This formulation allows the use of efficient dynamic programming techniques, such as *robust value iteration* (RVI) [28,41], which we adopt for solving these games.

This distinction corresponds to the difference between *agent first* and *nature first* semantics for robust partially observable MDPs in [7]. While in general this assumption can affect the game value, we establish in Theorem 1 that both semantics yield the same value in our setting (finitely-branching zero-sum ICSGs).

Theorem 1 (Player/nature-first Value Equivalence). *From any $s \in S$, $V_{\mathcal{G}}(s)$ is invariant under the player-first or nature-first semantics:*

$$\sup_{\sigma_1 \in \Sigma_1} \inf_{\sigma_2 \in \Sigma_2} \inf_{P \in \mathcal{P}} \mathbb{E}^{(\sigma_1,\sigma_2),P}_{\mathcal{G},s}[X] = \inf_{P \in \mathcal{P}} \sup_{\sigma_1 \in \Sigma_1} \inf_{\sigma_2 \in \Sigma_2} \mathbb{E}^{(\sigma_1,\sigma_2),P}_{\mathcal{G},s}[X].$$

Proof (Sketch). We prove the result top-down via construction of the *adversarial expansion* $\mathcal{G}^{\mathcal{A}}$ (Definition 10) and subsequent determinacy and value preservation results (Corollary 1) established in this section. Specifically, determinacy of the finite CSG $\mathcal{G}^{\mathcal{A}}$ justifies exchanging the order of optimisation between players and nature without changing the game value. Full proof in [24]. □

This value equivalence justifies using the nature-first semantics in implementations, so that nature's minimisation problem is solved only *once* per step, after which player strategies are derived from the minimising distributions P^*_{sa}. Henceforth, without loss of generality, we focus on the player-first semantics.

We next observe that optimal strategies for ICSGs with infinite-horizon objectives admit a memoryless form, which allows the game to be analysed via fixed-point equations over the state space.

Lemma 1 (Strategy class sufficient for optimality). *Given an* infinite-horizon *objective X for $\mathcal{G}$, each player has a* memoryless *robust optimal strategy, and nature has a* deterministic memoryless *optimal strategy.*

Proof (Sketch). Under (s, a)-rectangularity, nature's optimal resolution and players' continuation values depend only on the current state, so histories ending in the same state can be collapsed. Further, nature's independent choices across state–action pairs define a single transition function, so nature *deterministically* commits to one such function. Full proof in [24]. □

Henceforth in the zero-sum setting, we let Σ_i denote the set of *memoryless* strategies for each player $i \in \{1,2\}$, and interpret $\mathcal{P}$ as the set of transition functions resulting from *memoryless* nature strategies.

Using also the (s,a)-rectangularity of ICSGs, the *robust Bellman equation* for $\mathcal{G}$ (proved in [24]) is given by:

$$V(s) = \sup_{\sigma_1 \in \mathcal{D}(A_1(s))} \inf_{\sigma_2 \in \mathcal{D}(A_2(s))} \left\{ r_s^\sigma + \sum_{a \in A(s)} \sigma_{sa} \inf_{P_{sa} \in \mathcal{P}_{sa}} \sum_{s' \in S} P_{sas'} \cdot V(s') \right\} \quad (1)$$

where $\sigma_{sa} = \sigma_1(s,a_1)\sigma_2(s,a_2)$ with $a = (a_1,a_2)$ and $r_s^\sigma = \sum_{a \in A(s)} \sigma_{sa} r_{sa}$.

This characterises the fixed-point that our game solving algorithms will later compute iteratively. We remark that, unlike the standard Bellman equation for CSGs, Equation (1) includes nature's *inner problem* $\inf_{P_{sa} \in \mathcal{P}_{sa}}$, which captures transition uncertainty and is solved using a greedy algorithm adapted from the IMDP setting [41]; details are provided in [24]. Furthermore, whereas the Bellman equations for TSGs and RMDPs [28,41] optimise over *deterministic* player strategies by selecting pure actions, here we optimise over *randomised* (memoryless) strategies, i.e., distributions over actions. This reflects the added complexity of concurrent multi-agent interaction.

The CSG reduction. We now formally define the *adversarial expansion* $\mathcal{G}^\mathcal{A}$ for ICSG $\mathcal{G}$, which is a CSG containing intermediate states representing state–action pairs of $\mathcal{G}$. In the following, we use the operator $\cdot^\mathcal{A}$ to denote a structure associated with $\mathcal{G}^\mathcal{A}$. We write $\mathbb{V}(K)$ for the vertices of a polytope K and use notation such as "$s^\mathcal{A} = s \in S$" as shorthand for "$\exists s \in S.\, s^\mathcal{A} = s$".

Definition 10 (Adversarial expansion). *We define the* adversarial expansion *of ICSG $\mathcal{G}$ as a 2-player CSG $\mathcal{G}^\mathcal{A} = (\{1,2\}, S^\mathcal{A}, \bar{s}, A^\mathcal{A}, \Delta^\mathcal{A}, P^\mathcal{A})$ where:*

- $S^\mathcal{A} = S \cup S'$*, with* $S' = \{(s,a) \mid s \in S, a \in A\}$*;*
- $A^\mathcal{A} = (A_1^\mathcal{A} \cup \{\bot\}) \times (A_2^\mathcal{A} \cup \{\bot\})$*, with* $A_1^\mathcal{A} = A_1$*,* $A_2^\mathcal{A} = A_2 \cup \left(\bigcup_{s \in S, a \in A} \mathbb{V}[\mathcal{P}_{sa}] \right)$*;*
- $\Delta^\mathcal{A} : S^\mathcal{A} \to 2^{A_1^\mathcal{A} \cup A_2^\mathcal{A}}$*, such that if* $s^\mathcal{A} = s \in S$ *then* $\Delta^\mathcal{A}(s^\mathcal{A}) = \Delta(s)$*, else if* $s^\mathcal{A} = (s,a) \in S'$ *then* $\Delta^\mathcal{A}(s^\mathcal{A}) = \mathbb{V}[\mathcal{P}_{sa}]$*, else* $\Delta^\mathcal{A}(s^\mathcal{A}) = \emptyset$*;*
- $P^\mathcal{A} : S^\mathcal{A} \times A^\mathcal{A} \to \mathcal{D}(S^\mathcal{A})$ *such that if* $s^\mathcal{A} = s \in S \wedge s' = (s,a) \in S'$ *then* $P^\mathcal{A}(s^\mathcal{A}, a^\mathcal{A}, s') = 1$*, else if* $s^\mathcal{A} = (s,a) \in S' \wedge a^\mathcal{A} = (\bot, P_{sa}) \wedge s' \in S$ *then* $P^\mathcal{A}(s^\mathcal{A}, a^\mathcal{A}, s') = P_{sas'}$*, and* $P^\mathcal{A}(s^\mathcal{A}, a^\mathcal{A}, s') = 0$ *otherwise.*

As discussed earlier, player 2 in $\mathcal{G}^\mathcal{A}$ acts as a coalition of nature and player 2 in $\mathcal{G}$, such that the original player 2 acts at the S-states and nature moves at the S'-states. Given a choice $P_{sa} \in \mathcal{P}_{sa}$ of nature, a $\mathcal{G}$-transition $s \xrightarrow{a} s'$ corresponds to the two-step $\mathcal{G}^\mathcal{A}$-transition $s \xrightarrow{a} (s,a) \xrightarrow{(\bot, P_{sa})} s'$, and vice versa. In essence, the dynamics at S-states are unchanged. At an auxiliary state $(s,a) \in S'$, both players receive zero reward; and player 1 stays idle whilst player 2 deterministically selects a next-state distribution $P_{sa} \in \mathcal{P}_{sa}$, with P_{sa}^* being an optimal such choice as characterised by Lemma 1. The notion of adversarial expansion naturally extends to strategies, paths, rewards, and objectives (see [24]).

We highlight that $\mathcal{G}^{\mathcal{A}}$ is a finite-state, *finite-action* CSG. As formalised in [24], since an ICSG is polytopic, we can restrict player 2's actions to the vertices of the polytope $\mathcal{P}_{sa}$ at each S'-state without loss of optimality. The following results formalise the relationship between $\mathcal{G}$ and $\mathcal{G}^{\mathcal{A}}$.

Lemma 2 (Utility-preserving strategy bijection). *For any $\mathcal{G}$-profile $\sigma = (\sigma_1, \sigma_2)$, there exists a corresponding $\mathcal{G}^{\mathcal{A}}$-profile $\sigma^{\mathcal{A}} = (\sigma_1^{\mathcal{A}}, \sigma_2^{\mathcal{A}})$ and vice versa, such that $\inf_{P \in \mathcal{P}} u_1(\sigma, P) = u_1^{\mathcal{A}}(\sigma^{\mathcal{A}})$ and $\sup_{P \in \mathcal{P}} u_2(\sigma, P) = u_2^{\mathcal{A}}(\sigma^{\mathcal{A}})$, where $u_i^{\mathcal{A}}(\sigma^{\mathcal{A}})$ denotes player i's expected utility in $\mathcal{G}^{\mathcal{A}}$ under $\sigma^{\mathcal{A}}$.*

Proof. As shown in [24], $\mathcal{G}^{\mathcal{A}}$ preserves the set of possible paths, their probabilities and objective values. It follows directly that:

$$u_1(\sigma, P) = \sum_{\pi \in IPaths_{\mathcal{G},s}} \mathbb{P}^{\sigma}(\pi) \cdot X(\pi) = \sum_{\pi^{\mathcal{A}} \in IPaths_{\mathcal{G}^{\mathcal{A}},s}} \mathbb{P}^{\sigma}(\pi^{\mathcal{A}}) \cdot X^{\mathcal{A}}(\pi^{\mathcal{A}}) = u_1^{\mathcal{A}}(\sigma^{\mathcal{A}})$$

where $\mathbb{P}^{\sigma}$ is the path probability function under profile σ. Then by the zero-sum structure: $\sup_P u_2(\sigma, P) = u_2(\sigma, P^*) = -u_1(\sigma, P^*) = -u_1^{\mathcal{A}}(\sigma^{\mathcal{A}}) = u_2^{\mathcal{A}}(\sigma^{\mathcal{A}})$. □

Corollary 1 (Determinacy and Value Preservation). *$\mathcal{G}$ is determined iff $\mathcal{G}^{\mathcal{A}}$ is determined. Further, if both games are determined, then the robust value of $\mathcal{G}$ is equal to the value of $\mathcal{G}^{\mathcal{A}}$, i.e., $V_{\mathcal{G}}(s, X) = V_{\mathcal{G}^{\mathcal{A}}}(s, X^{\mathcal{A}})$.* (Proof in [24])

Theorem 2 ($\mathbf{RNE}_{\mathcal{G}} \Leftrightarrow \mathbf{NE}_{\mathcal{G}^{\mathcal{A}}}$). *In a determined zero-sum ICSG $\mathcal{G}$, for any $\mathcal{G}$-profile $\sigma \in \Sigma$, σ is an RNE in $\mathcal{G}$ with value $V_{\mathcal{G}}(s, X)$ iff $\sigma^{\mathcal{A}}$ is an NE in $\mathcal{G}^{\mathcal{A}}$ with value $V_{\mathcal{G}^{\mathcal{A}}}(s, X^{\mathcal{A}}) = V_{\mathcal{G}}(s, X)$.*

Proof (Sketch). Both directions follow from Definition 7 of RNE. The forward case additionally uses utility preservation (Lemma 2); the reverse relies on Definition 6 of the game value and value preservation (Corollary 1). Full proof in [24]. □

Solving zero-sum ICSGs. Finally, combining the above results, since $\mathcal{G}^{\mathcal{A}}$ is finite-state and finitely-branching, it is determined for all the objectives we consider [38]. By Corollary 1, the original game $\mathcal{G}$ is therefore also robustly determined with the same value. Moreover, since $\mathcal{G}$ is zero-sum, any RNE profile and its value coincides with an optimal profile and the game value.

Hence, by Theorem 2, we can perform robust verification of an ICSG $\mathcal{G}$ by solving the 2-player CSG $\mathcal{G}^{\mathcal{A}}$, e.g, with the value iteration approach from [34]. In fact, we do not need to explicitly construct $\mathcal{G}^{\mathcal{A}}$, nor its auxiliary states S' corresponding to the possible (s, a) pairs. Instead, for each state s, we first solve an inner optimisation problem over uncertainty set $\mathcal{P}_{sa}$ for each joint action a, and then solve a linear programming (LP) problem of size $|A|$ using the resulting values. We discuss this further in Section 6 and give full details in [24].

Finite-horizon properties. When X is a *bounded* probabilistic reachability or cumulative reward objective, the previous results still hold under two changes: 1) Robust optimal strategies are now *time-varying*, i.e., with the extended signature $\sigma : S \times H \rightharpoonup A$ and $P^* : S \times A \times H \rightharpoonup \mathcal{D}(S)$, where H represents *finite-memory* used to track the time-step. Accordingly, we extend $\mathcal{G}^{\mathcal{A}}$ with time-augmented states. 2) Finite-horizon objectives are evaluated over $k < \infty$ steps, thus *exact* game values can be computed via *robust backward induction* (RBI) [28,41], noting that $\mathcal{G}$ is always determined due to the finite game tree. We provide the detailed construction of $\mathcal{G}^{\mathcal{A}}$ in this setting in [24].

5 Nonzero-sum ICSGs

Next, we consider *nonzero-sum* 2-player ICSGs, where each player i maximises a distinct objective X_i, assuming for now that both objectives are infinite-horizon.[2] As in the zero-sum case, we continue to focus on player-first semantics and memoryless strategies, as both Theorem 1 and Lemma 1 extend to the nonzero-sum setting (see proofs in [24]).

We again reduce a 2-player ICSG $\mathcal{G}$ to its *adversarial expansion* $\mathcal{G}^{\mathcal{A}}$, which is a 3-player CSG, but in which the *nature* player now acts adversarially against both other players, aiming to minimise their social welfare. The reduction is more complex than the zero-sum case and requires an additional *filtering* step per iteration to identify robust equilibria. This reduction again allows us to build on standard CSG solution methods [34].

Our goal for nonzero-sum ICSGs is to find subgame-perfect ε-RNE (Definition 7), and more specifically, ε-RSWNE (Definition 8), which consider the *sum* of the utilities for the two players. We add a subscript $+$ to the relevant game notation (e.g., r, u, X, V) to indicate this. The *robust Bellman equation* is:

$$V_+(s) = \sup_{\sigma \in \Sigma_{\varepsilon\text{-RNE}}} \inf_{P \in \mathcal{P}} \mathbb{E}^{\sigma,P}_{\mathcal{G},s}[X_+] = \sup_{\sigma \in \Sigma_{\varepsilon\text{-RNE}}} \left[r_+(s,\sigma) + \sum_{a \in A} \inf_{P_{sa} \in \mathcal{P}_{sa}} f^{\sigma,P}_{sa} \right] \quad (2)$$

where $f^{\sigma,P}_{sa} := \sigma_{sa} \sum_{s' \in S} P_{sas'} V_+(s')$ and $\Sigma_{\varepsilon\text{-RNE}}$ denotes the set of *one-shot* ε-RNE. The second equality follows from a similar proof to the zero-sum case (see [24]). In this formulation, nature's inner problem is now to minimise the social welfare $u_+ := u_1 + u_2$, while each player $i \in \{1, 2\}$ maximises their individual expected payoff u_i. Consequently, we maximise over the set of ε-RNE, capturing equilibrium behaviour under worst-case uncertainty.

The adversarial expansion $\mathcal{G}^{\mathcal{A}}$ of ICSG $\mathcal{G}$ for the nonzero-sum case follows a similar construction to the zero-sum setting (Definition 10), but now models nature as an explicit third player distinct from player 2. Henceforth, if $a = (a_1, a_2)$, let $*a$ denote the flattened tuple a_1, a_2.

Definition 11 (Adversarial expansion). *We define the* adversarial expansion *of* $\mathcal{G}$ *as a* 3-player *CSG* $\mathcal{G}^{\mathcal{A}} = (N^{\mathcal{A}}, S^{\mathcal{A}}, \bar{s}, A^{\mathcal{A}}, \Delta^{\mathcal{A}}, P^{\mathcal{A}})$ *where:*

[2] Following the usual approach for nonzero-sum CSGs [34], we focus on ICSGs that can be seen as a variant of *stopping games* [14], where each player's target set is reached with probability 1 from all states under all profiles.

- $N^{\mathcal{A}} = \{1, 2, 3\}$, *with player 3 representing* nature*;*
- $S^{\mathcal{A}} = S \cup S'$, *with* $S' = \{(s, a) \mid s \in S, a \in A\}$*;*
- $A^{\mathcal{A}} = \times_{l=1}^{3}(A_l^{\mathcal{A}} \cup \{\bot\})$ *where* $A_1^{\mathcal{A}} = A_1$, $A_2^{\mathcal{A}} = A_2$ *and* $A_3^{\mathcal{A}} = \cup_{s \in S, a \in A} \mathbb{V}[\mathcal{P}_{sa}]$*;*
- $\Delta^{\mathcal{A}} : S^{\mathcal{A}} \to 2^{\cup_{i=1}^{3} A_i^{\mathcal{A}}}$, *such that if* $s^{\mathcal{A}} = s \in S$ *then* $\Delta^{\mathcal{A}}(s^{\mathcal{A}}) = \Delta(s)$*, else if* $s^{\mathcal{A}} = (s, a) \in S'$ *then* $\Delta^{\mathcal{A}}(s^{\mathcal{A}}) = \mathbb{V}[\mathcal{P}_{sa}]$ *and* $\emptyset$ *otherwise.*
- $P^{\mathcal{A}} : S^{\mathcal{A}} \times A^{\mathcal{A}} \to \mathcal{D}(S^{\mathcal{A}})$*, such that if* $s^{\mathcal{A}} = s \in S \wedge a^{\mathcal{A}} = (*a, \bot) \wedge s' = (s, a) \in S'$ *then* $P^{\mathcal{A}}(s^{\mathcal{A}}, a^{\mathcal{A}}, s') = 1$*, else if* $s^{\mathcal{A}} = (s, a) \in S' \wedge a^{\mathcal{A}} = (\bot, \bot, P_{sa}) \wedge s' \in S$ *then* $P^{\mathcal{A}}(s^{\mathcal{A}}, a^{\mathcal{A}}, s') = P_{sas'}$*, and* $P^{\mathcal{A}}(s^{\mathcal{A}}, a^{\mathcal{A}}, s') = 0$ *otherwise.*

Under this definition of $\mathcal{G}^{\mathcal{A}}$, a $\mathcal{G}$-transition $s \xrightarrow{(a_1,a_2)} s'$ corresponds to a two-step $\mathcal{G}^{\mathcal{A}}$-transition $s \xrightarrow{(a_1,a_2,\bot)} (s, (a_1, a_2)) \xrightarrow{(\bot,\bot,P_{s(a_1,a_2)})} s'$. Adversarial expansions of paths, strategies, rewards and objectives follow analogously to the zero-sum case, and are formalised in [24]. The following results relate $\mathcal{G}$ and $\mathcal{G}^{\mathcal{A}}$. Their proofs mirror the zero-sum case, with adaptations to the nonzero-sum definition of $\mathcal{G}^{\mathcal{A}}$.

Lemma 3 (Utility-preserving strategy bijection). *For any* $\mathcal{G}$*-profile* σ *under nature's choice of* $P \in \mathcal{P}$*, there exists a corresponding* $\mathcal{G}^{\mathcal{A}}$*-profile* $\sigma^{\mathcal{A},P}$ *and vice versa such that* $u_i(\sigma, P) = u_i^{\mathcal{A}}(\sigma^{\mathcal{A},P})$ *for* $i \in \{1, 2\}$*. Further, for any* $\sigma \in \Sigma$*, there exists a corresponding* $\sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}}$ *and vice versa such that* $\inf_{P \in \mathcal{P}} u_+(\sigma, P) = u_+^{\mathcal{A}}(\sigma^{\mathcal{A}}) = u_1^{\mathcal{A}}(\sigma^{\mathcal{A}}) + u_2^{\mathcal{A}}(\sigma^{\mathcal{A}})$*.*

Lemma 4 (ε-RNE$_{\mathcal{G}}$ $\Rightarrow$ ε-NE$_{\mathcal{G}^{\mathcal{A}}}$). *For any* $\mathcal{G}$*-profile* $\sigma \in \Sigma$*, if* σ *is an* ε*-RNE in* $\mathcal{G}$ *then* $\sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}}$ *is an* ε*-NE in* $\mathcal{G}^{\mathcal{A}}$*.* (Proof in [24])

Unlike the zero-sum setting, for Lemma 4 the converse does *not* necessarily hold: if $\sigma^{\mathcal{A}}$ is an ε-NE in $\mathcal{G}^{\mathcal{A}}$, σ need not be an ε-RNE in $\mathcal{G}$. This is because $\sigma_3^{\mathcal{A}}$ selects a transition function P^* that minimises the *total* utility of player 1 and 2, rather than each player's utility individually as required by the ε-RNE condition (see Definition 7). Therefore, any ICSG where nature's minimisation of the sum induces asymmetric incentives suffices as an example (see [24]).

Consequently, before identifying the RSWNE, for each state $s \in S$ we filter the set of ε-NE in $\mathcal{G}^{\mathcal{A}}$ to retain only those that correspond to an ε-RNE in $\mathcal{G}$, i.e., $\Sigma_{\varepsilon\text{-RNE}}^{\mathcal{A}} := \{\sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}} \mid \sigma \in \Sigma_{\varepsilon\text{-RNE}}\}$. Note that filtering is applied separately at each state since we construct subgame-perfect equilibria.

Filtering $\Sigma_{\varepsilon\text{-NE}}^{\mathcal{A}}$ for $\Sigma_{\varepsilon\text{-RNE}}$. Our method of filtering is based on a notion of *deviations* made by players. In this section, we fix a state $s \in S$ and candidate ε-RNE profile $\sigma \in \Sigma$. We designate player i as the *deviator*, whose strategy deviations $\sigma_i' \in \Sigma_i$ from σ will be evaluated. Further, we define the *deviation gain* of player i under its deviation σ_i' and nature's choice of $P \in \mathcal{P}$ as:

$$u_i^{\Delta}(\sigma_i', P) := u_i(\sigma_{-i}[\sigma_i'], P) - u_i(\sigma, P). \tag{3}$$

Lemma 5 (ε-RNE condition over pure deviations). *Let* Σ_i^{det} *denote the set of (memoryless) deterministic strategies for player* $i \in N = \{1, 2\}$*. A profile* $\sigma \in \Sigma$ *is an* ε*-RNE iff the following condition holds:*

$$\overline{V}_{i,\sigma} := \sup_{P \in \mathcal{P}} \sup_{\sigma_i' \in \Sigma_i^{det}} u_i^{\Delta}(\sigma_i', P) \ \leq \varepsilon \quad \forall i \in N.$$

Proof (Sketch). The ε-RNE condition for σ (Definition 7) can be rewritten as $\sup_P \sup_{\sigma'_i \in \Sigma_i} u_i^{\Delta}(\sigma'_i, P) \leq \varepsilon$. Since the expected utility (from any state s) is linear in the deviation σ'_i, the maximal gain is attained by a deterministic deviation. So it suffices to consider pure deviations. Full proof in [24]. $\square$

Observe that $\overline{V}_{i,\sigma}$ corresponds exactly to the *optimistic value* of an IMDP $\mathcal{G}^{\mathcal{D}}_{i,\sigma}$ in which player i acts as the agent. We refer to this IMDP as the *deviation IMDP* and formalise it in [24]. The value correspondence relies on the IMDP's construction whereby, given state space $S^{\mathcal{A}} = S \cup S'$ of $\mathcal{G}^{\mathcal{A}}$, transitions from states in S to S' depend exclusively on the fixed player's strategy, whilst those from S' to S are governed by nature's choice of $P \in \mathcal{P}$. Further, the reward assigned to each state $s \in S$ corresponds to the expected reward gain at s if player i deviates from σ_i to σ'_i.

Therefore, following Lemma 5, if the deviation IMDP has optimistic value $\overline{V}_{i,\sigma} \leq \varepsilon$ for both players $i \in \{1, 2\}$, then the candidate profile σ constitutes an ε-RNE profile of $\mathcal{G}$. We can thus characterise our goal in solving $\mathcal{G}$ as identifying:

$$\Sigma_{\varepsilon\text{-RNE}} = \left\{ \sigma \in \Sigma \;\middle|\; \sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}}_{\varepsilon\text{-NE}} \;\wedge\; \forall i \in \{1, 2\}.\, \overline{V}_{i,\sigma} \leq \varepsilon \right\}. \tag{4}$$

Solving nonzero-sum ICSGs. Altogether, a profile $\sigma \in \Sigma$ is an ε-RSWNE in $\mathcal{G}$ iff its corresponding $\sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}}_{\varepsilon\text{-RNE}}$ maximises $u^{\mathcal{A}}_{+}$ in $\mathcal{G}^{\mathcal{A}}$. Hence, computing ε-RSWNE in the 2-player ICSG $\mathcal{G}$ reduces to finding SWNE in the 3-player CSG $\mathcal{G}^{\mathcal{A}}$ over $\Sigma^{\mathcal{A}}_{\varepsilon\text{-RNE}}$. While this would in principle require a general 3-player CSG solver (e.g., [33]), such algorithms rely on nonlinear programming and are computationally expensive. However, by exploiting the zero-sum coalitional structure of $\mathcal{G}^{\mathcal{A}}$, the *trimatrix* game at each state $s \in S$ can be reduced to a *bimatrix* game (see [24]). This can thus be solved more efficiently using the 2-player nonzero-sum CSG solution approach from [34], together with the inner-problem solution algorithm in [24] to ensure robustness. As in the zero-sum case, this computation does not require explicit construction of $\mathcal{G}^{\mathcal{A}}$.

Once we have the set of one-shot ε-NE of $\mathcal{G}^{\mathcal{A}}$ at each $s \in S$, we filter for the ε-RNE equivalents: for each profile $\sigma^{\mathcal{A}} \in \Sigma^{\mathcal{A}}_{\varepsilon\text{-NE}}$, we: 1) compute $\overline{V}_{i,\sigma}$ for each player $i \in \{1, 2\}$ on the deviation IMDP $\mathcal{G}^{\mathcal{D}}_{i,\sigma}$; and 2) retain σ if all $\overline{V}_{i,\sigma} \leq \varepsilon$. The resulting profiles correspond to the ε-RNE in $\mathcal{G}$, from which we select the one maximising u_+. This gives an ε-RSWNE profile and values in $\mathcal{G}$ (by Lemma 3).

Finite- and mixed-horizon properties. Since players' objectives are distinct, they may differ in time horizon. If both X_1 and X_2 are finite-horizon, then we define the analysis horizon as the maximum of the two, i.e., $k := \max(k_1, k_2)$. In mixed-horizon cases, where one objective is finite-horizon and the other infinite-horizon, we transform the game into an equivalent one with two infinite-horizon objectives on an augmented model, following [32]. Thus, we focus on cases where both objectives are either finite- or infinite-horizon.

The infinite-horizon framework generalises to finite-horizon objectives in a similar way to the zero-sum setting. Additionally, we consider exact RNE and RSWNE (i.e., $\varepsilon = 0$), and account for potentially different player horizons by labelling time-augmented states that record when each player's target is reached within their respective horizon. The full construction is detailed in [24].

6 Value Computation for Two-player ICSGs

Sections 4 and 5 have presented the theoretical foundations for solving zero-sum and non-zero sum ICSGs, respectively, and described how a reduction to an *adversarial expansion* CSG provides the basis for iterative solution methods. In this section, we present some additional implementation details and discuss correctness and complexity. Full details are provided in [24].

Both approaches use elements of robust dynamic programming for RMDPs, i.e, robust value iteration (RVI) for infinite-horizon properties and robust backward induction (RBI) for finite-horizon properties, and of value iteration based methods for CSGs [34]. For both zero-sum and nonzero-sum objectives, the procedure performed per iteration for each state $s \in S$ is:

Algorithm 1 RVI/RBI update for state $s \in S$ in $\mathcal{G} = (N, S, \bar{s}, A, \Delta, \check{P}, \hat{P})$

```
1: for all a ∈ A(s) do
2:     P*_sa ← SolveInnerProblem(s, a, V_prev, P̌, P̂)        ▷ see [24]
3: end for
4: Z ← ConstructNFG(P*_s, V_prev)                             ▷ see [24]
5: V_next[s] ← SolveNFG(Z)
```

The SolveInnerProblem function (line 2) uses an algorithm adapted from a greedy method for IMDPs [41], which we detail in [24]. At line 4, we build a normal form game: for zero-sum ICSGs this a *matrix* game, reusing the zero-sum CSG algorithms in [34]; for nonzero-sum ICSGs, we build a (general-sum) *bimatrix* game using multi-player CSG algorithms in [33], which coincide with those for nonzero-sum 2-player CSGs in [34]. Our derivations appear in [24], where we demonstrate how to directly compute values of $\mathcal{G}$ without explicit construction of $\mathcal{G}^{\mathcal{A}}$.

At line 5, SolveNFG computes the matrix game value via an LP formulation [40, 52] in the zero-sum case. For nonzero-sum ICSGs, this is a multi-step procedure which involves: 1) enumerating NE for bimatrix games, using e.g., the *Lemke-Howson algorithm* [36]; and 2) filtering these for RNE as outlined in Section 5 using deviation IMDPs. The latter can be done using IMDP verification algorithms already supported in PRISM-games [31]. If no profiles remain, our algorithm terminates early; otherwise the value of state s is updated to the RSWNE value of $\mathcal{Z}$.

Correctness. This relies on the reduction of solving an ICSG $\mathcal{G}$ to solving a standard CSG $\mathcal{G}^{\mathcal{A}}$, whose correctness is established in Section 4 (zero-sum) and 5 (nonzero-sum). Correctness of the underlying CSG algorithm over $\mathcal{G}^{\mathcal{A}}$ is inherited from [33, 34], which relies on classical results, e.g., correctness of value iteration and backward induction [44] and solution of matrix games [40, 52]. Robustness is ensured by solving the inner problem, for which the correctness of our adapted algorithm in [24] follows from [41].

Complexity. Runtime depends on the number of iterations and per-iteration cost, both of which are dependent on game size. Finite-horizon objectives require exactly k iterations, while infinite-horizon ones iterate until convergence, which may be exponential in $|A|$ in the worst case [11] (even for MDPs [23]). Since line 2 takes $O(|S| \log |S|)$ time per execution, the per-iteration cost for zero-sum ICSGs is $O\left(|S|^2|A| \log |S| + |S| L_{|A|}\right)$, where $L_{|A|}$ is the cost of solving an LP problem of size $|A|$. This is polynomial in $|S|$ and $|A|$ with e.g., Karmarkar's algorithm [29]; or exponential under the simplex algorithm [16], which is PSPACE-complete in the worst case [20] but performs well on average [51]. For nonzero-sum ICSGs, each iteration takes worst-case exponential time due to NE enumeration [2].

Strategy synthesis. At each state, solving the NFG yields both the value(s) and optimal player strategies, while nature's optimal strategy is given by the returned values of SolveInnerProblem (line 2 of Algorithm 1). These are memoryless and taken from the final RVI iteration for infinite-horizon objectives; but time-varying and taken from each RBI step for finite-horizon objectives.

7 Tool support and Experimentation

We extended PRISM-games [31] to support modelling and solution of 2-player ICSGs, building on its existing functionality for CSGs and IMDPs. The tool and case studies are available at [25].

Experimental setup. We evaluate the efficiency and scalability of our techniques, comparing, as a point of reference, to standard (non-robust) CSG solution from PRISM-games.[3] We use the benchmarks from [34], obtaining ICSGs by perturbing all non-0/1 probability CSG transitions with a two-sided uncertainty $\pm\epsilon$. This includes a combination of finite-, infinite- and mixed-horizon properties specified in the logic rPATL [13,34]. All experiments were run on a 3.2 GHz Apple M1 with 16 GB memory. Further statistics for benchmark models and an extended set of results can be found in [24].

Q1. How does verification time for ICSGs compare to CSGs? Results for solving CSGs and ICSGs under adversarial uncertainty, completed within a 2-hour time limit, are shown in Table 1 for zero-sum games and Table 2 for nonzero-sum games. Overall, the increase in verification time when moving from CSGs to ICSGs is significantly more pronounced in the nonzero-sum setting, highlighting the added complexity introduced by the nonzero-sum formulation.

Across all benchmarks, verification times for zero-sum ICSGs remained within a factor of two of their CSG counterparts (see Table 1), with all test instances (except for *User-centric network* with $K > 4$) solved within 2 hours. These include models with over 2 million states and 10 million transitions. By contrast, for nonzero-sum ICSGs, verification completes within 2 hours for models up to 0.4 million states and 1.3 million transitions. This aligns with the underlying reduction framework: in the zero-sum setting, the problem reduces to a 2-player game, whereas in the nonzero-sum case it is a more complex 3-player game.

[3] Due to improvements in PRISM-games, some statistics differ slightly from [34].

Table 1: Zero-sum verification results, with full statistics in [24].

Case study: [params], ϵ Property	Param. values	Avg # actions	States	Val. Iters CSG	Val. Iters ICSG	Verif. time (s) CSG	Verif. time (s) ICSG	Value CSG	Value ICSG
Robot coordination: $[l]$, 0.01 $\langle\langle rbt_1\rangle\rangle \mathtt{R}_{\min=?}[\mathtt{F}\ g_1]$	4	2.07,2.07	226	19	18	0.15	0.27	4.55	4.39
	8	2.52,2.52	3,970	29	29	2.50	3.61	8.89	8.63
	12	2.68,2.68	20,450	39	37	16.22	31.73	13.15	12.84
User centric network: $[K]$, 0.01; $\langle\langle usr\rangle\rangle \mathtt{R}_{\min=?}[\mathtt{F}\ f]$	3	2.11,1.91	32,214	60	59	789.66	834.92	0.04	0.03
	4	2.31,1.92	104,897	81	81	3525.67	3729.40	4.00	4.00
Aloha: $[b_{\max}]$, 1/257 $\langle\langle u_2, u_3\rangle\rangle \mathtt{R}_{\min=?}[\mathtt{F}\ s_{2,3}]$	2	1.00120,1.00274	14,230	105	103	5.31	5.61	4.34	4.28
	3	1.00023,1.00054	72,566	128	125	18.50	26.39	4.54	4.46
	4	1.00004,1.00009	413,035	195	190	225.69	291.72	4.62	4.53
	5	1.00001,1.00002	2,237,981	343	327	4669.54	4260.59	4.65	4.54
Jamming radio systems: $[chans, slots]$, 0.01 $\langle\langle u\rangle\rangle \mathtt{P}_{\max=?}[\mathtt{F}\ (sent \geq slots/2)]$	4,6	2.17,2.17	531	7	7	0.32	0.46	0.84	0.80
	4,12	2.49,2.49	1,623	13	13	1.39	2.94	0.77	0.71
	6,6	2.17,2.17	531	7	7	0.25	0.45	0.84	0.80
	6,12	2.49,2.49	1,623	13	13	1.46	2.37	0.77	0.71

Table 2: Nonzero-sum verification results, with full statistics in [24].

Case study: [params], ϵ Property	Param. values	Avg # actions	States	Val. Iters CSG	Val. Iters ICSG	Verif. time (s) CSG	Verif. time (s) ICSG	Value CSG	Value ICSG
Robot coordination: $[l,k]$, 0.01 $\langle\langle r_1 : r_2\rangle\rangle_{\max=?}\ (\mathtt{P}[\neg c\ \mathtt{U}^{\leq k} g_1] + \mathtt{P}[\neg c\ \mathtt{U}^{\leq k} g_2])$	4,4	2.07,2.07	226	4	4	0.26	0.60	1.55	1.50
	8,8	2.52,2.52	3,970	8	8	1.03	54.74	0.92	0.84
	12,12	2.68,2.68	20,450	12	12	8.55	2895.60	0.49	0.40
Robot coordination: $[l,k]$, 0.01 $\langle\langle r_1 : r_2\rangle\rangle_{\max=?}\ (\mathtt{P}[\neg c\ \mathtt{U}^{\leq k} g_1] + \mathtt{P}[\neg c\ \mathtt{U}\ g_2])$	4,8	2.10,2.04	226	14	14	1.22	17.54	2.00	2.00
	4,16	2.12,2.05	3,970	14	11	2.08	75.23	2.00	2.00
Aloha (deadline): $[b_{\max}, D]$, 1/257 $\langle\langle u_1 : u_2, u_3\rangle\rangle_{\max=?}\ (\mathtt{P}[\mathtt{F}\ s_1] + \mathtt{P}[\mathtt{F}\ s_{2,3}])$	1,8	1.0048,1.0111	14,230	23	23	0.45	5.13	1.99	1.99
	2,8	1.0012,1.0027	72,566	23	23	1.17	107.63	1.98	1.97
	3,8	1.0002,1.0005	413,035	22	22	3.96	3306.24	1.97	1.97
Medium access: $[e_{\max}, k_1, k_2]$, 0.01 $\langle\langle p_1 : p_2, p_3\rangle\rangle_{\max=?}\ (\mathtt{R}[\mathtt{C}^{\leq k_1}] + \mathtt{R}[\mathtt{C}^{\leq k_2}])$	10,20,25	1.91,3.63	10,591	25	25	577.60	614.46	26.10	25.88
	15,20,25	1.94,3.75	33,886	25	25	1148.02	6109.14	34.35	34.06

While the main overhead arises from solving the inner problem at each iteration, in the zero-sum setting this is often offset by faster RVI convergence. For example, in the *Aloha* model with $b_{\max} = 5$, ICSG verification outperformed CSGs in runtime and required noticeably fewer iterations to converge. Additionally, consistent with [34], for both zero- and nonzero-sum CSGs and ICSGs, verification time appears to depend more on the number of actions per player/coalition than the number of states. For example, the minimally branched *Aloha* instances (averaging close to one action per coalition) are verified relatively efficiently compared to other games with similarly sized state spaces.

Q2. How does ϵ affect verification time? Interestingly, as shown in Figure 1, verification times are often the lowest when ϵ is very small or close to its maximum value, while intermediate ϵ values tend to be slower. This non-monotonic behaviour again reflects the trade-off due to uncertainty: larger ϵ increases the per-iteration cost by giving nature more choices, but also accelerates convergence by flattening the value landscape [41], thus reducing the number of iterations required until convergence. Thus ϵ can be tuned to balance robustness and efficiency. However, the net effect of ϵ on verification time is model-specific: e.g., in the *Robot Coordination* case with $l = 12$ from Table 1, ICSG verification required fewer iterations but still resulted in a longer overall verification time.

Q3. How does ϵ influence the computed value? Increasing ϵ yields more conservative results, as shown in Figure 2. This is to be expected, as expanding the

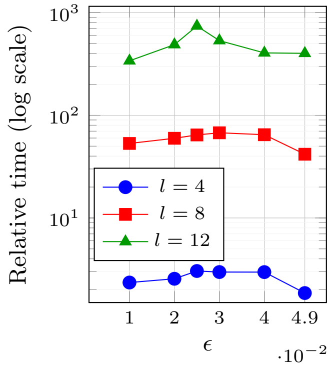

Fig. 1: Total verification time relative to CSG baseline in the first nonzero-sum *Robot coordination* case study, which requires $\epsilon < 0.05$.

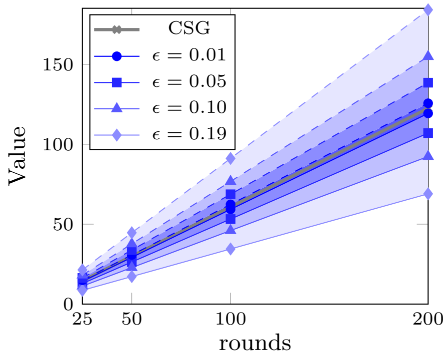

Fig. 2: ICSG values over game size in the *Intrusion Detection* case study, under two resolutions of uncertainty: *adversarial* (solid lines) and *controlled* by coalition 1 (dashed).

uncertainty set enables nature to select "worse" transitions that reduce the game value. The effect is amplified in larger, more connected models, in which pessimistic transitions propagate over longer paths [54]. Figure 2 also illustrates a practical use of ICSG verification in computing two-sided, ϵ-parametrised bounds on verification results, which can be interpreted as confidence intervals under transition uncertainty. This is useful in safety-critical and performance-sensitive settings, where robustness must be ensured against worst-case security attacks, while enabling estimation of, e.g., optimistic operational performance.

8 Conclusion

We have introduced robust CSGs, an extension of classical CSGs with transition uncertainty that enables principled analysis of multi-agent, concurrent stochastic systems with imprecise dynamics. Focusing on interval uncertainty, i.e., ICSGs, we developed verification algorithms for the 2-player setting, for both finite- and infinite-horizon objectives. Our approach relies on a value-preserving reduction of ICSGs to standard CSGs, thereby allowing reuse of elements of the solution methods for both CSGs and RMDPs, plus custom adaptations and filtering. Our implementation in PRISM-games shows that solution in the zero-sum case scales comparably to standard CSGs, with runtime increases below a factor of two. In the nonzero-sum case, computational demands are higher but we still scale successfully to large CSGs. Future work could explore RCSGs with richer uncertainty models and/or alternative solution concepts (e.g., robust correlated equilibria), and consider objectives with more general temporal specifications.

Data Availability Statement. The models, tools, and scripts to reproduce our experimental evaluation are archived and available at [25].

Acknowledgments. Supported by the EPSRC Centre for Doctoral Training no. EP/Y035070/1 and the UKRI AI Hub on Mathematical Foundations of AI.

References

1. Aghassi, M., Bertsimas, D.: Robust game theory. Mathematical programming **107(1)**, 231–273 (2006)
2. Avis, D., Rosenberg, G.D., Savani, R., Von Stengel, B.: Enumeration of nash equilibria for two-player games. Economic theory **42(1)**, 9–37 (2010)
3. Badings, T., Simao, T.D., Suilen, M., Jansen, N.: Decision-making under uncertainty: beyond probabilities: Challenges and perspectives. International Journal on Software Tools for Technology Transfer **25(3)**, 375–391 (2023)
4. Bellman, R.: Dynamic programming. science **153(3731)**, 34–37 (1966)
5. Berthon, R., Katoen, J.P., Mittelmann, M., Murano, A.: Robust strategies for stochastic multi-agent systems. In: Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems. p. 2437–2439. AAMAS '25, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2025)
6. Bouyer, P., Markey, N., Stan, D.: Mixed nash equilibria in concurrent terminal-reward games. In: FSTTCS 2014 (2014)
7. Bovy, E.M., Suilen, M., Junges, S., Jansen, N.: Imprecise probabilities meet partial observability: Game semantics for robust POMDPs. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24. pp. 6697–6706 (8 2024), main Track
8. Castro, P.F., D'Argenio, P.: Polytopal stochastic games. In: Principles of Verification: Cycling the Probabilistic Landscape: Essays Dedicated to Joost-Pieter Katoen on the Occasion of His 60th Birthday, pp. 99—-117. Springer (2024)
9. Chatterjee, K., De Alfaro, L., Henzinger, T.A.: Strategy improvement for concurrent reachability and turn-based stochastic safety games. Journal of computer and system sciences **79(5)**, 640–657 (2013)
10. Chatterjee, K., Goharshady, E.K., Karrabi, M., Novotný, P., Žikelić, u.: Solving long-run average reward robust MDPs via stochastic games. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence. IJCAI '24 (2024)
11. Chatterjee, K., Henzinger, T.A.: A survey of stochastic ω-regular games. Journal of Computer and System Sciences **78(2)**, 394–413 (2012)
12. Chatterjee, K., Sen, K., Henzinger, T.A.: Model-checking ω-regular properties of interval Markov chains. In: International Conference on Foundations of Software Science and Computational Structures. pp. 302–317. Springer (2008)
13. Chen, T., Forejt, V., Kwiatkowska, M., Parker, D., Simaitis, A.: Automatic verification of competitive stochastic systems. Formal Methods in System Design **43**, 61–92 (2013)
14. Chen, T., Han, T., Kwiatkowska, M.: On the complexity of model checking interval-valued discrete time Markov chains. Information Processing Letters **113(7)**, 210–216 (2013)
15. Condon, A.: The complexity of stochastic games. Information and Computation **96**(2), 203–224 (1992)
16. Dantzig, G.B.: Programming of interdependent activities: Ii mathematical model. Econometrica, Journal of the Econometric Society pp. 200–211 (1949)
17. De Alfaro, L., Henzinger, T.A., Kupferman, O.: Concurrent reachability games. Theoretical computer science **386(3)**, 188–217 (2007)
18. De Alfaro, L., Majumdar, R.: Quantitative solution of omega-regular games. In: Proceedings of the thirty-third annual ACM symposium on Theory of computing. pp. 675–683 (2001)

19. Farhat, Z.U., Ghosh, D., Atia, G.K., Wang, Y.: Online robust multi-agent reinforcement learning under model uncertainties (2025), https://arxiv.org/abs/2508.02948
20. Fearnley, J., Savani, R.: The complexity of the simplex method. In: Proceedings of the forty-seventh annual ACM symposium on Theory of computing. pp. 201–208 (2015)
21. Filar, J., Vrieze, K.: Competitive Markov decision processes. Springer Science & Business Media (2012)
22. Givan, R., Leach, S., Dean, T.: Bounded-parameter Markov decision processes. Artificial Intelligence **122(1-2)**, 71–109 (2000)
23. Hansen, K.A., Ibsen-Jensen, R., Miltersen, P.B.: The complexity of solving reachability games using value and strategy iteration. In: International Computer Science Symposium in Russia. pp. 77–90. Springer (2011)
24. He, A.Y., Parker, D.: Robust verification of concurrent stochastic games under interval uncertainty. Extended Version (2026), https://arxiv.org/abs/2601.12003, extended version of the TACAS 2026 paper
25. He, A.Y., Parker, D.: Artifact for "Robust verification of concurrent stochastic games" (2026). https://doi.org/10.5281/zenodo.18189280
26. Ho, C.P., Petrik, M., Wiesemann, W.: Fast bellman updates for robust MDPs. In: International Conference on Machine Learning. pp. 1979–1988. PMLR (2018)
27. Howard, R.A.: Dynamic Programming and Markov Processes. John Wiley & Sons, New York (1960)
28. Iyengar, G.N.: Robust dynamic programming. Mathematics of Operations Research **30(2)**, 257–280 (2005)
29. Karmarkar, N.: A new polynomial-time algorithm for linear programming. In: Proceedings of the sixteenth annual ACM symposium on Theory of computing. pp. 302–311 (1984)
30. Klibano, P.: Uncertainty, decision, and normal form games,". Journal of Economic Theory, forthcoming (1993)
31. Kwiatkowska, M., Norman, G., Parker, D., Santos, G.: PRISM-games 3.0: Stochastic game verification with concurrency, equilibria and time. In: Proc. 32nd International Conference on Computer Aided Verification (CAV'20). LNCS, vol. 12225, pp. 475–487. Springer (2020)
32. Kwiatkowska, M., Norman, G., Parker, D., Santos, G.: Equilibria-based probabilistic model checking for concurrent stochastic games. In: International Symposium on Formal Methods. pp. 298–315. Springer (2019)
33. Kwiatkowska, M., Norman, G., Parker, D., Santos, G.: Multi-player equilibria verification for concurrent stochastic games. In: International Conference on Quantitative Evaluation of Systems. pp. 74–95. Springer (2020)
34. Kwiatkowska, M., Norman, G., Parker, D., Santos, G.: Automatic verification of concurrent stochastic systems. Formal Methods in System Design **58(1)**, 188–250 (2021)
35. Kwiatkowska, M., Norman, G., Parker, D., Santos, G.: Correlated equilibria and fairness in concurrent stochastic games. In: International Conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 60–78. Springer (2022)
36. Lemke, C.E., Howson, Jr, J.T.: Equilibrium points of bimatrix games. Journal of the Society for industrial and Applied Mathematics **12(2)**, 413–423 (1964)
37. Littman, M.L.: Markov games as a framework for multi-agent reinforcement learning. In: Machine learning proceedings 1994, pp. 157–163. Elsevier (1994)
38. Martin, D.A.: The determinacy of blackwell games. The Journal of Symbolic Logic **63(4)**, 1565–1581 (1998)

39. Meggendorfer, T., Weininger, M., Wienhöft, P.: Solving robust Markov decision processes: Generic, reliable, efficient. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39(25), pp. 26631–26641 (2025)
40. v. Neumann, J.: Zur theorie der gesellschaftsspiele. Mathematische annalen **100(1)**, 295–320 (1928)
41. Nilim, A., El Ghaoui, L.: Robust control of Markov decision processes with uncertain transition matrices. Operations Research **53(5)**, 780–798 (2005)
42. Osborne, M.J., et al.: An introduction to game theory, vol. 3(3). Springer (2004)
43. Perchet, V.: Finding robust nash equilibria. In: Algorithmic Learning Theory. pp. 725–751. PMLR (2020)
44. Raghavan, T., Filar, J.A.: Algorithms for stochastic games—a survey. Zeitschrift für Operations Research **35(6)**, 437–472 (1991)
45. Roch, Z., Wang, Y.: Distributionally robust Markov games with average reward (2025), https://arxiv.org/abs/2508.03136
46. Roughgarden, T.: Algorithmic game theory. Communications of the ACM **53(7)**, 78–86 (2010)
47. Shapley, L.S.: Stochastic games. Proceedings of the national academy of sciences **39(10)**, 1095–1100 (1953)
48. Shi, L., Mazumdar, E., Chi, Y., Wierman, A.: Sample-efficient robust multi-agent reinforcement learning in the face of environmental uncertainty. arXiv preprint arXiv:2404.18909 (2024)
49. Strehl, A.L., Littman, M.L.: An empirical evaluation of interval estimation for Markov decision processes. In: 16th IEEE International Conference on Tools with Artificial Intelligence. pp. 128–135. IEEE (2004)
50. Strehl, A.L., Littman, M.L.: A theoretical analysis of model-based interval estimation. In: Raedt, L.D., Wrobel, S. (eds.) Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005). pp. 856–863. ACM (2005)
51. Todd, M.J.: The many facets of linear programming. Mathematical programming **91(3)**, 417–436 (2002)
52. Von Neumann, J., Morgenstern, O.: Theory of games and economic behavior, princeton (1944)
53. Wang, Y., Velasquez, A., Atia, G., Prater-Bennette, A., Zou, S.: Robust average-reward Markov decision processes. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37(12), pp. 15215–15223 (2023)
54. Wiesemann, W., Kuhn, D., Rustem, B.: Robust Markov decision processes. Mathematics of Operations Research **38(1)**, 153–183 (2013)